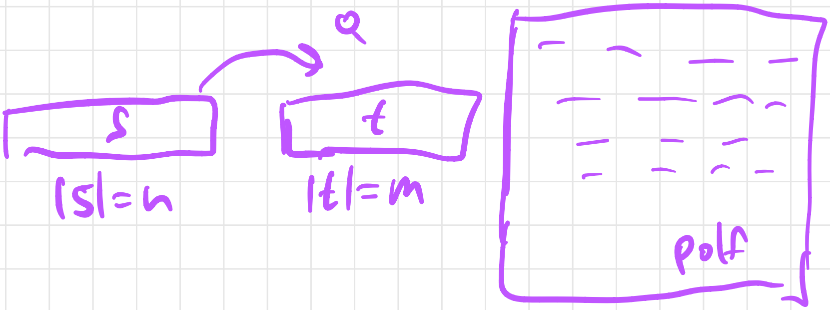
Strings

+Био

Поиск подстроки в строке



Z-функции

π-функции

- простые
- $O(n+m)$

Boyer-Moore

Boyer-Moore-Horspool

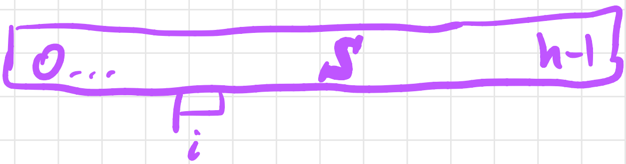
Knuth-Morris-Pratt

Rabin-Karp

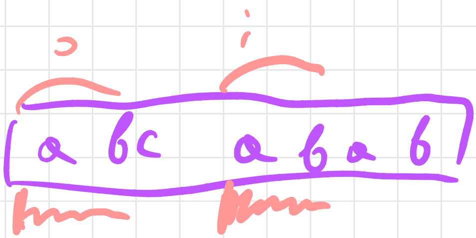
Z-функция

Main, Lorentz 1984

Gusfield



$$z[i] = \max k : s[0..k-1] == s[i..i+k-1]$$



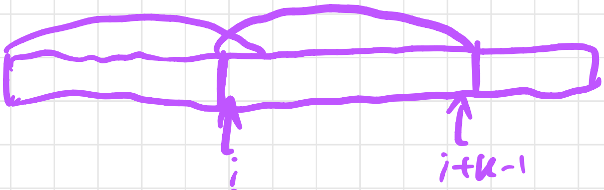
$z[i] = 2$

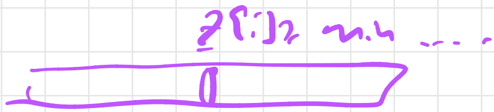
$O(n^2)$

Применение к поиску подстроки

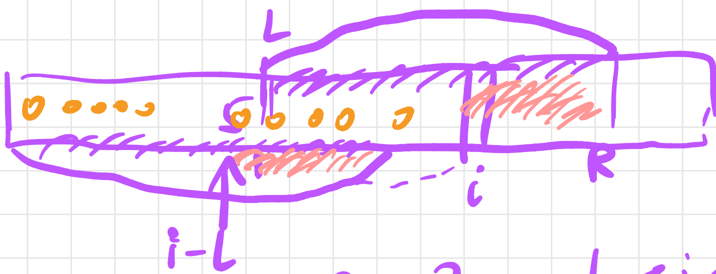
$z(i) \geq |s|$

Вычисление z-функции



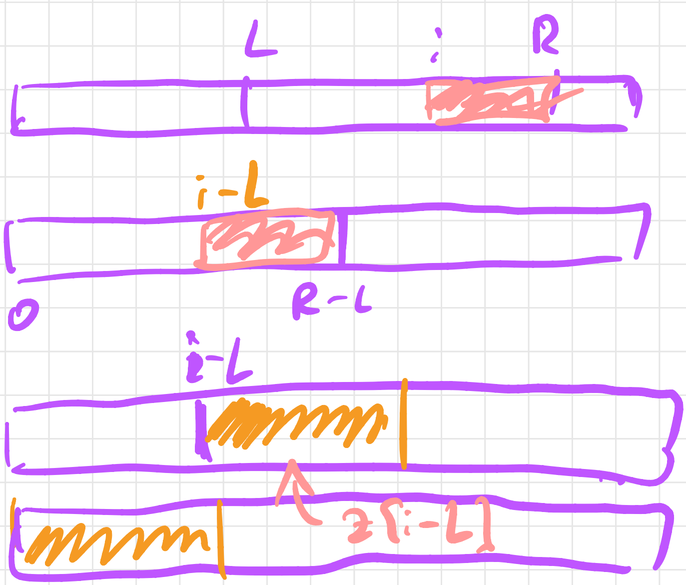



```
calc z(S):
  z = [0 ..... 0]
  z[0] = |S|
  L = 0
  R = 0
  for i := 1 ... |S|-1
    k = 0
    if i ≤ R:
      k = min(z[i-L], R-i+1)
    while i+k ≠ |S| and
                  S[k] == S[i+k]:
      k += 1
    z[i] = k
    if i + z[i] - 1 > R
      L = i
      R = i + z[i] - 1
```



$z[i] = ?$ $\quad L \le i \le R$

$$\Rightarrow z[i] \ge \min\left(z[i-L],\ R-i+1\right)$$



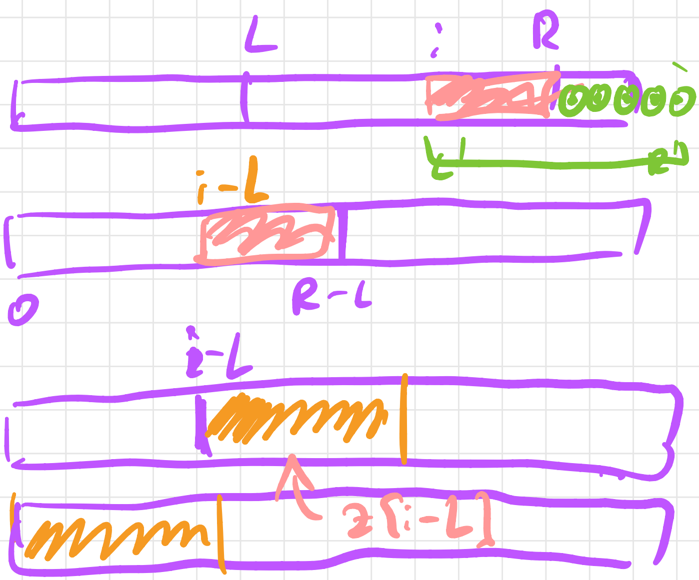
a b a b a a a

7 0 3 0 1 1 1

Доказательство.

① $O(n)$

$z[i] \geq \min(z[i-L], R-i+1)$

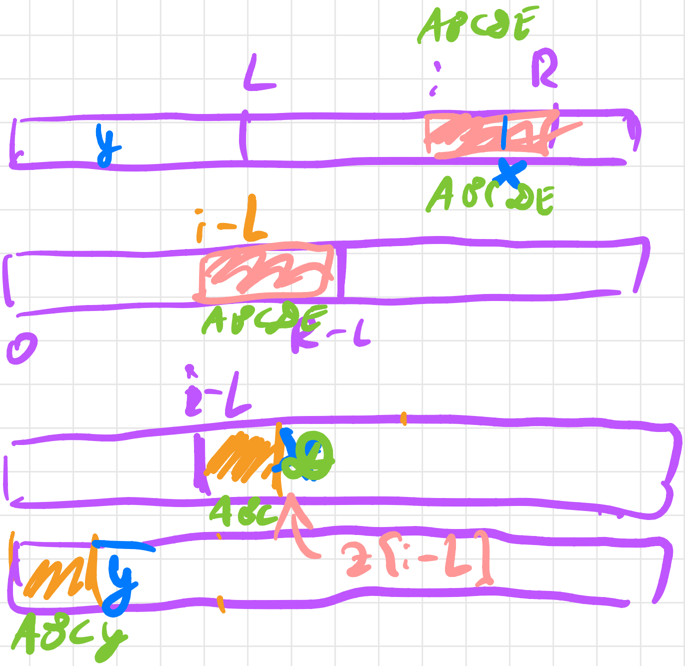


$z[i] = R - i + 1$

$z[i]$++++

$$t = R' - R + 1$$

$z[i] = \min$



$t = O(1)$

(ABCy ABCDE ABCDE)

π-функция
(префикс)

Knuth-Morris-Pratt

a b a c a b a
0 0 1 0 1 2 3

u

a b a c a b
2

π[i]:

S_0 ... S_i

k k

max k:
k < вся строка

s: a b a c

π(abac) ≠ 4
π(a) ≠ 1

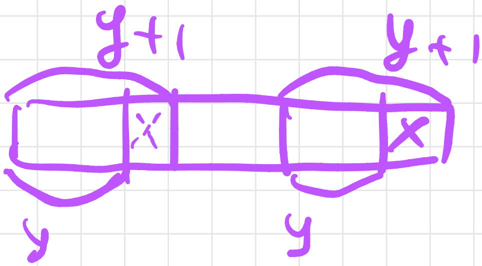
a b a c a b a | x

π = 0 0 1 0 1 2 3 ?

$\overline{\text{aба}}\,c\,\overline{\text{aба}}\,x$ $\qquad x = ?$

4 символа ($x = c$)

< 4 символов (иначе)



$y \in ???$

$S = \overline{\text{aб}}\text{aca}\overline{\text{бa}}|$ $\qquad$ $\underline{\text{aб}}\text{aca}\underline{\text{б}}$

$y \in \{3, 1, 0\} = A$

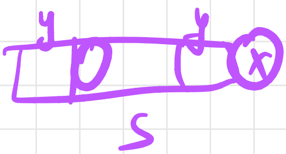
↑ мн-во таких длин, что они совпадают

$$A = \{ y \mid S_{0..y-1} == S_{|S|-y..|S|-1} \}$$

$$\pi(Sx) = \begin{cases} S_{MAX(A)} == x \Rightarrow MAX(A)+1 \\ ? \end{cases}$$

$$\pi(Sx) = \max_{\substack{y \in A \\ S_y = x}} y+1$$

$$A = \{0, 1, 3\}$$

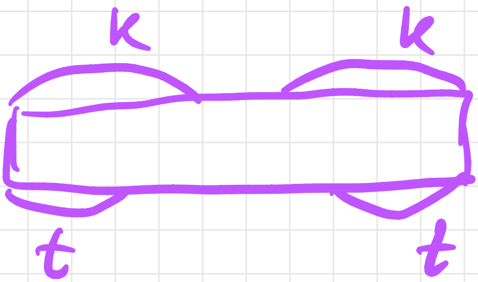


$\pi(abacaba) = 3 \rightarrow S_y = x$ ✓

$\downarrow\ y' < MAX(A)$
$\in A.$

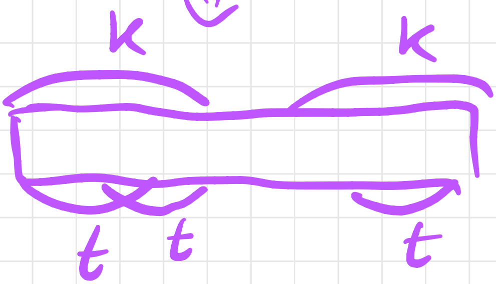
бордер

2-ой по величине бордер



t — бордер

$\Downarrow$



Def k-бордер, если $S_{0..k-1} ==$
$== S_{|S|-k...|S|-1}$

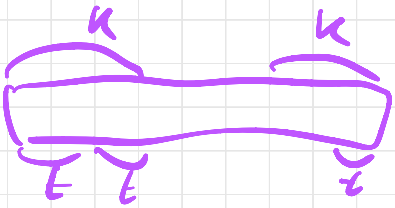
k — макс-бордер

Утв t — бордер ($t<k$)

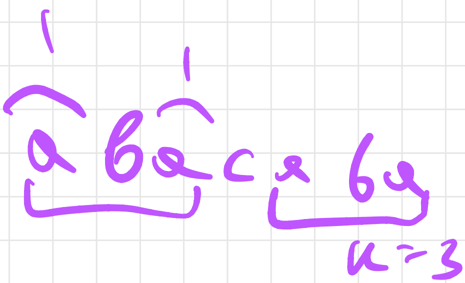
$\Rightarrow t$ — бордер строки $S_0..S_{k-1}$

Утв
k – макс бордер
t – бордер $s_0 \dots s_{k-1}$

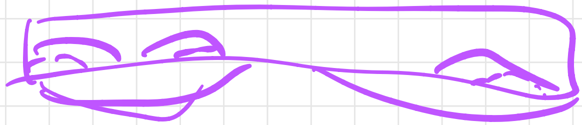
$\Rightarrow$ t – бордер s.



Утв: k – макс бордер
$\Rightarrow$ второй бордер это макс бордер $s_0 \dots s_{k-1}$



$A = \{0, 1, 3)$



A: $A \in \pi(s) = k$

$A \in \pi(s_0 \dots s_{k-1}) = k'$

$A \in \pi(s_0 \dots s_{k'-1}) \dots$

$s = abacaba$ $\quad y = \pi(s) = 3$

$\pi(sx) \to s_3 = x\ ? \Rightarrow \pi = 4$

$\pi(sx) \to y' = \pi(s_0 \dots s_2)$

$A = \{0, 1, 3\}$

$s_{y'} = x\ ?$ $\quad \pi \Rightarrow y' + 1$

? ?

a b a c a b a | b a a

0 0 1 0 1 2 3 | 2 3 1

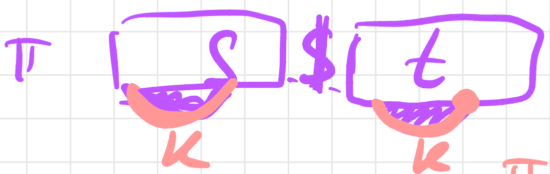
$\pi(s)$:

```
res = [0 ...   0]        // res[0] = 0.

for i = 1 ... |s|-1:
    k = res[i-1];
    while k ≠ -1:
        if s_i == s_k:
            break
        else
            if k == 0:
                k = -1
            else
                k = res[k-1].
    res[i] = k+1
```

1) применения

$$s \xrightarrow{Q} t$$



vector<char>

$\pi_i == |s| \to$ вхождение

2) Время работы $O(n)$

Лм: $\pi(sx) \le \pi(s)+1$

? ?

a b a c a b a | b a a

0 0 1 0 1 2 3 | 2 3 1

$t=1$

$\pi(s) \to \pi(sx) = \pi(s)+1$

$\pi(s) \to \pi(sx) < \pi(s)+1$

увеличений k в коде $\le n$

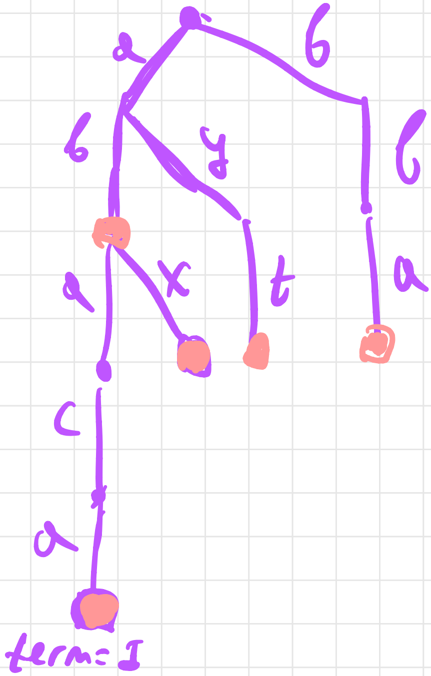
падений вниз можно $\le n$

Суммарное время работы $\le 2n$

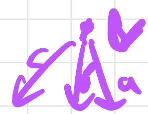
$k = \pi($тек. префикс$)$

Trie (Бор)

abaca
abx
ayt
bba
ab



Node {
go: char → Node
term: bool
# Term. в вершине
}




```
class Node:
    def __init__(self):
        self.term = False
        self.go = dict()
```

```
root = Node()

def add(s):                |s|
    cur = root
    for c in s:
        if not c in cur.go:
            cur.go[c] = Node()
        cur = cur.go[c]
    cur.term = true
```

- Множество (строками)

add(s) ←

contains(s) ← $\Theta(|s|)$

remove(s) ←